Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations 1. Thesis and Dissertation Collection, all items

1968-12

# On primal integer programming methods.

Schottle, Howard Thomas

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/40096

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

*Downloaded from NPS Archive: Calhoun*

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

# UNITED STATES
# NAVAL POSTGRADUATE SCHOOL

# THESIS

## ON PRIMAL INTEGER PROGRAMMING METHODS

by

Howard Thomas Schottle

December 1968

Thesis
S342

*This document has been approved for public release and sale; its distribution is unlimited.*

# ON PRIMAL INTEGER PROGRAMMING METHODS

by

Howard Thomas Schottle
Lieutenant, United States Navy
B.S., United States Naval Academy, 1961

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
December 1968

Signature of Author Howard T Schottle

Approved by Harold Greenberg
Thesis Advisor

Chairman, Department of Operations Analysis

R. F. Rinehart
Academic Dean

[illegible]

# ABSTRACT

Advances in the field of integer programming have recently been made by Young and Glover, who have developed primal all integer integer programming algorithms. It is the purpose of this paper to gain computational experience and suggest techniques to improve the efficiency of the algorithm. A first phase procedure is developed to obtain an initial feasible integer solution.

LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIF. 93940

# TABLE OF CONTENTS

| Section | | Page |
|---|---|---|
| I | Introduction | 7 |
| II | Description of Algorithm | 9 |
| III | Discussion of Problem | 16 |
| IV | Computational Results | 20 |
| V | Discussion of Results | 25 |
| BIBLIOGRAPHY | | 28 |
| APPENDIX | | 29 |
| | A - An Example Problem | 29 |
| | B - Fortran IV Program of Algorithm | 37 |

## LIST OF ILLUSTRATIONS

Figure Page

1. Characteristics of the Test Problems 21

2. Number of Iterations to Solve the Test Problems 22

## I Introduction

Advances in the field of integer programming have recently been made by Young[10] and Glover[5], who have developed primal all inter-integer programming algorithms.

A primal algorithm is one that proceeds through a series of primal feasible bases towards an optimal solution of the problem. The two principle advantages of a primal method are:

1) at any point during the solution, the problem may be stopped and a feasible solution exists that may be used as a "good" approximation:

2) a feasible solution arrived at by any method may be used as a starting point for a primal algorithm. The most famous example of a primal programming algorithm is Dantzig's Simplex Algorithm.[2,3]

Until Young developed his first primal algorithm[9] there were no finite primal algorithms available in the literature; however, there was one privately attributed to Gomory. Algorithms developed prior to this time were either cutting plane or enumeration types. Young's first algorithm was extremely complex and difficult to implement for any practical purpose. Subsequently Young simplified his algorithm[10].

The objective here is to gain computational experience with the new algorithms of Young and Glover and to suggest techniques that will improve their efficiency.

The two algorithms are not significantly different, thus they will be considered as one where the minor differences will be noted.

The algorithm contains nondeterministic rules which will be investigated to determine their effect on the progress of the algorithm. The rule for the selection of a pivot column in the Simplex Algorithm for solving linear programs is an example of a nondeterministic rule. In its most general form the rule is to select any column that has a negative coefficient in the objective function. The user has to determine which column he will select as a pivot column from a set of columns that have negative coefficients in the objective function.

The algorithms require a knowledge of a feasible integer solution. In a problem this may be difficult to obtain . A first phase procedure is developed to obtain an initial feasible integer solution.

## II Description of the Algorithm

The basic problem under consideration is:

find $x_j$, $j=1,2,\ldots,n$ that maximizes $x_o$

where

$$x_o + \sum_{j=1}^{n} a_{oj}\bar{x}_j = a_{oo}$$

$$x_i + \sum_{j=1}^{n} a_{ij}\bar{x}_j = a_{io} \qquad i=1,2,\ldots,m$$

$$x_{m+j} - \bar{x}_j = 0 \qquad j=1,2,\ldots,n \tag{1}$$

$$x_j \geq 0 \text{ and integer} \qquad j=1,2,\ldots,n$$

Without loss of generality it can be assumed that all $a_{ij}$ are integers and all $a_{io} \geq 0$. If not, appropriate techniques are available to insure this.

In this section the procedures to solve (1) are described and set forth. The method is contained in a primal algorithm that relies heavily on Dantzig's Simplex Algorithm [2,3] and Gomory's All Integer Integer Programming Algorithm[7].

Essentially the algorithm selects a pivot column and a source row which is used to generate a Gomory cut[6]. The cut equation is then adjoined to the problem and used as the pivot row for a change of basis. Each pivot does not necessarily result in an increase in the objective function and more restrictive rules are necessary for the selection of a pivot column and source row to insure that the procedure is finite.

Prior to proceeding to the description of the algorithm it is necessary to define the following terms and notational conventions.

Let

$v$ = index of pivot column

$s$ = index of source row

$J$ = set of indicies of non basic variables

$\theta_j = \min (a_{io}/a_{ij})$ for $a_{ij} > 0$ for $j$ in $J$.

$[t]$ = greatest integer less than or equal to $t$.

$S(j)$ = set of row indices $i$ such that

$$0 \leq \left[ a_{io} / a_{ij} \right] \leq \theta_j \quad \text{for } j \text{ in } J.$$

$A_j$ = jth column of coefficients for $j$ in $J$.

The cut equation is defined as

$$\sum_{j \in J} \left[ a_{sj} / a_{sv} \right] \overline{x}_j + u = \left[ a_{so} / a_{sv} \right] \qquad (2)$$

where $u$ = slack variable associated with the cut equation

A transition cycle is a pivot that results in an increase in the objective function. A stationary cycle is a pivot that does not increase the objective function.

The reference constraint is a constraint of the form

$$\sum_{j \in J} a_{rj}\bar{x}_j \leq a_{ro} \tag{3}$$

that is adjoined to the problem to guide the progress of the algorithm through a sequence of stationary cycles. This is necessary to insure that the number of intervening stationary cycles between two transition cycles is finite which is necessary to insure that the algorithm will terminate in a finite number of steps.

The reference vector is defined as

$$\bar{A}_j = A_j/a_{rj} = (a_{oj}/a_{rj}, a_{ij}/a_{rj}, \ldots , a_{mj}/a_{rj}) \tag{4}$$

for all j such that $a_{rj} > 0$. $\bar{A}_j$ is not defined for $a_{rj} \leq 0$.

The exact form of the reference constraint is not specified and is one of the choices of the user. The reference constraint must satisfy the following.

$$\begin{aligned} &1) \quad A_j \overset{L}{<} 0 \quad \text{implies} \quad a_{rj} > 0^{*} \\ &2) \quad a_{rj} < 0 \quad \text{implies} \quad \bar{A}_j \overset{L}{>} \bar{A}_v \end{aligned} \tag{5}$$

---

* The symbol $\overset{L}{<}$ means lexicographically less than. A vector x is said to be lexicographically less than a vector y if the first non zero component of the vector (x-y) is negative.

Young suggests using (3) as a reference constraint where all $a_{rj} = 1$ and $a_{ro}$ is a positive constant selected to be large enough so that no feasible solutions are excluded from the problem. Glover suggests using a surrogate constraint as the reference constraint.

A surrogate constraint is "an inequality implied by the constraints of an integer program, and designed to capture useful information that cannot be extracted from the parent constraints individually but is nevertheless a consequence of their conjunction".[5] In general a surrogate constraint is of the form of equation 8 below and is obtained from a linear combination of the constraints of the original problem.

If the original problem (1) is written in the form:

$$\begin{aligned} &\text{find } x = (x_1, x_2, \ldots x_n)^T \quad \text{that} \\ &\text{maximizes} \quad x_o = cx \\ &\text{where} \quad Ax \leq b \\ &\text{and} \quad x \geq 0 \ \text{ and integer} \end{aligned} \tag{6}$$

when $c = (a_{01}, a_{02}, \ldots, a_{0n})$, $A = (a_{ij})$, $i = 1,2,\ldots m$,
$j = 1,2,\ldots n$

and $b = (a_{10}, a_{20}, \ldots a_{m0})^T$. If the integer constraint is dropped, the dual of this problem is then:

$$\begin{aligned} &\text{find} \quad w = (w_1, w_2, \ldots w_m) \\ &\text{that minimizes} \quad w_0 = w_b \\ &\text{where} \quad wA \geq c \\ &\qquad\quad\ \ w \geq 0 \end{aligned} \tag{7}$$

The reference constraint as suggested by Glover [4] is then defined as

$$\sum_{j=1}^{n} a_{rj} x_j \leq a_{ro} \qquad (8)$$

where $a_{rj} = w^* A_j$

$a_{ro} = w^* b$

$w^*$ = a feasible solution to the dual.

An appropriate source row selection rule is defined as any rule which guarantees, for any row k, that at finite intervals a tableau occurs in which $a_{kv} \leq a_{ko}$. The rule is necessary to insure that a transition cycle will occur at finite intervals.

The following rules completely determine the algorithm:

1) Adjoin a reference constraint to the problem.
2) Check for optimality. The current tableau is optimal if $a_{oj} \geq 0$ for $j \epsilon J$ or if $a_{ov}/a_{rv} > -1/a_{ro}$.
3) Determine the pivot column v, such that $\bar{A}_v \overset{L}{<} \bar{A}_j$ for $j \neq v$ and $a_{rj} > 0$.
4) Select a source row, s, from S(j) by an appropriate source row selection procedure.
5) Adjoin to the tableau the Gomory cut (2) derived from the source row.
6) Carry out the usual simplex change of basis with v as the pivot column and the Gomory cut as the pivot row.

7) Delete the pivot column and pivot row from the system.

8) Go to 2.

Steps 5,6, and 7 may be combined to one step as:

5a) Execute the change of basis as follows:

1) $\bar{a}_{ij} = - a_{ij}$ for $j = v$

2) $\bar{a}_{ij} = a_{ij} - \left[a_{sj}/a_{sv}\right] a_{is}$ for $j \neq v$ (9)

Step 5a is equivalent to steps 5,6, and 7.

The above procedure always selects the column with the lexicographic minimum $\bar{A}_j$ as the pivot column. It may be desirable to execute a transition cycle when possible in order to immediately increase the objective value and possibly decrease the number of iterations. In this case, Rules 3 and 8 above are replaced by:

3a. If for some j in J and $a_{oj} < 0$ $\theta_j \geq 1$, designate that column as the pivot column. If none exists use rule 3 above.

8a. Go to 1 if a transition cycle has been executed. Go to 2 otherwise.

This procedure requires that the reference constraint be revised after each transition cycle since the problem is essentially restarted as if it were a new problem. Because of this, computationally it would be advantageous to have an easily determinable reference constraint.

Optimality is achieved if dual feasibility occurs, that is if all $a_{oj} \geq 0$, or if $a_{ov}/a_{rv} > -1/a_{ro}$. The second criteria occurs if no positive integer values exist for the non basic variables that will satisfy the reference constraint and improve the solution. A complete proof is contained in[4].

Rules 3 and 4 are the essential rules necessary to insure that the algorithm is finite. Basically, these two rules insure that $\bar{A}_v^{(k)}$, the reference vector for the $k^{th}$ iteration, is lexicographically greater than $\bar{A}_v^{(k-1)}$, the reference vector for the previous iteration. Since the first component of $\bar{A}_v$ is $a_{ov}/a_{rv}$ and $\bar{A}_v \overset{L}{<} \bar{A}_j$ for $j \neq v$ and $j$ in $J$, for each iteration, then when $\bar{A}_v \overset{L}{>} 0$, the problem is at optimality.

## III Discussion of Problem

Within the algorithm there are three non deterministic rules, the form of the reference constraint, the source row selection rule, and whether to execute a transition cycle as soon as possible or to select the column with the lexicographically minimum reference vector.

The reference constraint must satisfy the criteria of (5.1) and (5.2). One such constraint is that of (3). Denote this by Reference Constraint One (RC1). The critical point is determining $a_{ro}$. Any large positive integer can be used that does not exclude any feasible integer solution. It is possible to determine the upper bound on the summation of the variables by solving the linear program of (6) where the integer constraint is dropped, $c = (1,1,\ldots 1)$ and all other variables are as defined in (6).

A second form of the reference constraint is that of (8), a surrogate constraint, as suggested by Glover[4]. This will be denoted by Reference Constraint Two (RC2).

The final form of the reference constraint considered is:

$$cx \leq c_o \qquad (10)$$

where cx is the original objective function and $c_o$ is some large integer constraint such that no feasible integer solutions are excluded from the problem. Denote this form as Reference Constraint Three (RC3). As in RC1, $c_o$ can be obtained from the solution of the linear program (6) without the integer constraints.

If a transition cycle is executed out of order, that is when first possible rather than always selecting that column with the lexicographically minimum reference vector, then the reference constraint must be revised after each transition cycle. Because of this RC2 was not considered when transition cycles were executed out of order. It did not appear to be a reasonable procedure to solve the dual problem after each transition cycle.

To meet the criteria of the definition of an appropriate source row selection rule, it is necessary to insure that for any row i only a finite number of cycles will exist such that $a_{iv} \geq a_{io}$. In order for this to occur, once a row has been selected such that $a_{iv} \geq a_{io}$, it is necessary to repeatedly select this row until it no longer qualifies as a source row.

The source row selection rules (SRR) that were used are

SRR1) a) Select that row in S(v) with the smallest row index number.
  b) Repeatedly select this row until it no longer qualifies.
  c) go to a

SRR2) a) Generate a sequential list of row indicies.
  b) Select the first index of the list that is in S(v).
  c) same as SRR1b.

d) Continue down the list until a row is found that is in S(v), go to c.

e) If none found before end of list go to b.

SRR3) a) Generate the vector $G_i$ for each row in S(v) where

$$g(j) = \left[a_{ij}/a_{iv}\right] \quad j = 1,2,\ldots n \qquad (11)$$

b) Select source row s such that $G_s \overset{L}{>} G_i$ for all i in S(v) $i \neq s$

c) same as SRR1b

d) Go to a

SRR1 is used because of the computational simplicity. SRR2 is one of the rules Young suggested[10]. SRR3 is used to increase the values of the coefficients of the objective function. As a justification for this, consider rule 5a of the algorithm, which defines the pivoting process for the objective function row,

$$\overline{a}_{oj} = a_{oj} - \left[a_{sj}/a_{sv}\right] a_{ov} \qquad j \neq v \qquad (12)$$

It can be seen that $a_{oj}$ will be changed by an amount $-g(j)\ a_{ov}$ in the next tableau. Since $a_{ov} < 0$, it is desired to have $G_s$ as large as possible (in some sense), in order that $a_{oj}$ will increase towards the optimality criteria of all $a_{oj} \geq 0$.

With the above choices, fifteen different combinations were considered on twelve test problems[8]. The next section describes the problems and the results.

The algorithm is demonstrated by means of a sample problem in Appendix 2.

## IV Computational Results

The results of the various choices discussed in the last section will be presented here. The characteristics of the twelve test problems are summarized in Figure 1. Problems 7 and 8 are identical to 5 and 6 respectively with two constraints omitted. The constraints are 0-1 bounds on two variables. For problems 9, 10, and 11, RC2, the reference constraint derived from a surrogate constraint, was exactly the same as RC3, the reference constraint that uses the objective function as a constraint.

Figure 2 is the summary of the results of the problems. RC3 with transition cycles executed as soon as possible was considered and a few problems were run but the results were so poor that it was not pursued any further. The reason for the poor results was that the optimality criteria of $a_{ov}/a_{rv} \geq -1/a_{ro}$ was never met. This resulted because $a_{ov}/a_{rv}$ was always equal to -1 and $-1/a_{ro}$ was a negative fraction. This made it necessary to achieve dual feasibility, before the problem could be terminated, which considerably increased the number of interations.

The computational procedures are straight forward and are completely described by the rules in Section three. One variation considered was that of obtaining an initial basic feasible solution to the problem when one was not readily available.

| Problem Number | Number of Constraints | Number of Variables | Type* of Constraint | Denominator ** |
|---|---|---|---|---|
| 1 | 4 | 5 | all LE | 183 |
| 2 | 4 | 5 | all LE | 258 |
| 3 | 4 | 5 | all LE | 320 |
| 4 | 4 | 5 | all LE | 205 |
| 5 | 6 | 5 | all LE | 19400 |
| 6 | 6 | 5 | all LE | 32000 |
| 7 | 4 | 5 | all LE | 19400 |
| 8 | 4 | 5 | all LE | 32000 |
| 9 | 6 | 6 | all LE | 2000 |
| 10 | 7 | 7 | all GE | 32 |
| 11 | 7 | 7 | all GE | 32 |
| 12 | 3 | 4 | all GE | 72 |

Characteristics of the Test Problems

* LE indicates that the constraints are less than or equal to inequalities.

GT indicates that the constraints are greater than or equal to inequalities.

** The denominator is the determinate of the coefficients of the basic variables when the problem is solved as a linear program. It gives a general indication as to the difficulty of finding an integer solution.

Figure One

| RC | 1* | | | 2* | | | 3* | | | 1** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SRR | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 |
| 1 | 32 | 67 | 77 | 854 | 401 | 387 | 168 | 316 | 287 | 391 | 303 | 29 |
| 2 | 24 | 59 | 24 | 422 | 1021 | 355 | 115 | 569 | 188 | 515 | 419 | 28 |
| 3 | 284 | 537 | 171 | 479 | 1124 | 480 | 339 | 1202 | 520 | 29 | 29 | 29 |
| 4 | 5 | 5 | 5 | 399 | 1111 | 325 | 623 | 691 | 48 | 39 | 46 | 29 |
| 5 | 29651 | 90k | 26533 | 90k | 90k | 31514 | 90k | 90k | 90k | 46876 | 47669 | 348 |
| 6 | 38522 | 90k | 90822 | 90k | 90k | 90k | 90k | 90k | 90k | 234 | 234 | 173 |
| 7 | 53319 | 90k | 90k | 90k | 90k | 203 | 90k | 90k | 90k | 54075 | 53867 | 348 |
| 8 | 90k | 90k | 90k | 90k | 90k | 90k | 90k | 90k | 90k | 234 | 234 | 173 |
| 9 | 180 | 600 | 240 | 12289 | 19757 | 2548 | 12289 | 18757 | 2548 | 3684 | 4936 | 246 |
| 10 | 19 | 18 | 18 | 31 | 20 | 24 | 39 | 20 | 24 | 18 | 33 | 18 |
| 11 | 44 | 43 | 25 | 34 | 120 | 39 | 34 | 120 | 39 | 19 | 22 | 21 |
| 12 | 2197 | 11798 | 3822 | 226 | 165 | 116 | 39 | 39 | 37 | 21 | 451 | 195 |

Number of Iterations to Solve the Test Problems.

* Pivot column always the column with the lexicographically minimum reference vector.

** Transition cycles executed when possible.

A k indicates that the problem had not terminated in the indicated number of iterations (x$10^3$).

Figure Two

This occurred in problems 10, 10, and 12. A Phase I technique analogous to that of the Simplex Agorithm was used. Rule 5a was used in place of rules 5, 6, and 7 of the algorithm for the computations. This made it necessary to employ a special procedure to eliminate artificial variables from the problem during Phase I. At various points during Phase I, equation exist of the form

$$-u_j + y_i = 0 \tag{13}$$

where $u_j$ is a slack variable generated from a cut and $y_i$ is the $i^{th}$ artificial variable. When equations of this form appear during Phase I they are dropped and the $j^{th}$ column is deleted from the problem. This is done because the artificial variable, $y_i$, equals zero and the equation implies $u_j$ equals zero. It is then eliminated from all other equations in the problem. One further special procedure was employed during Phase I. When w is reduced to zero, it is possible that some of the artificial variables are still in the problem. They must be deleted before proceeding to Phase II. To accomplish this a pivot is carried out on any element that will give the form of (13). This can best be accomplished by pivoting on an element where some $y_i$ is the basic variable and the coefficient is equal to one. Any column may be selected as the pivot column if the pivot is carried out in an equation that has an articial as a basic variable. This is possible because $a_{io} = 0$ for all rows i that have artifical variables as basic variables (when w = 0). This special procedure is carried out until equationsof the form of 13 are obtained and are then deleted from the system. When all artificals have been removed an initial basic feasible solution has been found and the problem can be solved by the basic algorithm.

## V Discussion of Results

From the summary of results in Figure 2 a number of conclusions can be made. It appears that the best procedure investigated is to execute a transition cycle when first possible in order to reduce the number of iterations. The best reference constraint, of those investigated, was RCl as originally suggested by Young[10]. This constraint produced the least number of iterations in general. It also was the easiest to obtain and use in terms of computational simplicity. This result was somewhat unexpected. It was initially believed [4] that RC2, the surrogate constraint, would yield better results due to the nature of a surrogate constraint. This advantage however was completely overshadowed by the advantages gained by executing a transition cycle as soon as possible. RC2 was not investigated when a transition cycle is executed out of order. It does not appear to be a reasonable procedure to solve the dual problem after each transition cycle, as would be required.

The results of the various choices of source row selection rules were not as clear as that of the reference constraints. The only rule that significantly decreased the number of iterations was SRR3 when used in conjunction with RCl, and when transition cycles are executed out of order. For the other cases it appears that SRR-3 is slightly better than the other two rules.

All of the problems tested here were of moderate size yet the number of iterations necessary to reach optimality was usually quite large. On larger problems the number of iterations may be so large that the algorithm would be of no practical use, even when the most ingenious and efficient techniques are used.

One of the most promising aspects of the algorithm is that it may open the way for a variety of composite or specialized algorithms to fit the needs of specialized users.

Among the areas that need future investigation, the following appear to be most significant:

1) More work is necessary to reduce the number of iterations necessary to reach optimality. On a number of the test problems, optimality was achieved in relatively few iterations, but the criteria was not met for many more iterations. On test problems 1 and 2 optimality was achieved after one iteration but even in the best cases at least 20 or more iterations were necessary to recognize optimality. Glover has made an outstanding contribution toward this end with his modified optimality criteria[4] as used here.

2) Another area, which requires future work is devising a method that requires less computer storage and more importantly, fewer calculations. The basic tableau requires a space of (m + n + 2) x (n + 1). It would not take a very large problem to exhaust the memory of a present day computer. If the necessary number of calculations could be

decreased, a larger number of iterations would be tolerable. In test problems 5 and 6 it took approximately 10 minutes of execution time to carry out 90,000 iterations.

3) Further work is also necessary to determine an initial basic feasible solution. It has been shown here that a Phase I technique can be used successfully.

The algorithm is a major step forward in the field of integer programming. In theory, it provides a finite algorithm that will solve any integer programming problem. In practice the number of iteration necessary to reach optimality can be large, especially on the more difficult problems if the choices for the non-deterministic rules are not made carefully.

# BIBLIOGRAPHY

1. BALINSKI, M. L., "Integer Programming Methods, Uses, Computations", Management Sci. 12, 253-313 (1965).

2. DANTZIG, George B., "Maximization of a Linear Function of Variables Subject to Linear Inequalities", in T. C. Koopmans (ed.), Activity Analysis of Production and Allocation, Cowles-Commission Monograph No.13, Wiley, New York, N. Y., 1951.

3. DANTZIG, George B., Linear Programming and Extensions, Princeton University Press, Princeton, New Jersey,1963.

4. GLOVER, Fred, "A New Foundation for a Simplified Primal Integer Programming Algorithm," Opns. Res., 16, 727-740 (1968).

5. GLOVER, Fred, "Surrogate Constraints" Opns. Res., 16, 741-749 (1968).

6. GOMORY, Ralph E., "An Algorithm for Integer Solutions to Linear Programs", in Robert L. GRAVES and Philip WOLFE (eds.), Recent Advances in Mathematical Programming, pp. 269-302, McGraw-Hill, in New York, N.Y.,1963.

7. GOMORY, Ralph E., "An All-Integer Programming Algorithm", J. F. MUTH and Gerald L. THOMPSON (eds.) Industrial Scheduling, pp. 193-206, Prentice-Hall, Englewood Cliffs, New Jersey, 1963.

8. HALDI, John, "25 Integer Programming Test Problems", Working Paper No. 43, Graduate School of Business, Stanford University, December 1964.

9. YOUNG, R. D., "A Primal (All-Integer) Integer Programming Algorithm," J. Res. National Bureau of Standard-B, Mathematics and Mathematical Physics, Vol. 69B, No. 3, July-September 1965.

10. YOUNG, R. D., "A Simplified Primal (All Integer) Integer Programming Algorithm, Opns. Res., 16, 750-782 (1968.

## Appendix A

The following problem will be worked as an example of the algorithm.

Find $x = (x_1, x_2, x_3, x_4, x_5)$ that

maximizes $x_o = x_3 + x_4 + x_5$

$$
\begin{aligned}
\text{where}\quad & 2\bar{x}_1 + 3\bar{x}_2 + \bar{x}_3 + 2\bar{x}_4 + 2\bar{x}_5 + y_1 &&= 18\\
& 3\bar{x}_1 + 2\bar{x}_2 + 2\bar{x}_3 + \bar{x}_4 + 2\bar{x}_5 \quad + y_2 &&= 15\\
& -6\bar{x}_1 \quad + \bar{x}_3 \qquad + y_3 &&= 0\\
& -7\bar{x}_2 \quad + \bar{x}_4 \qquad + y_4 &&= 0\\
& -\bar{x}_1 \qquad + x_1 &&= 0\\
& -\bar{x}_2 \qquad + x_2 &&= 0\\
& -\bar{x}_3 \qquad + x_3 &&= 0\\
& -\bar{x}_4 \qquad + x_4 &&= 0\\
& -\bar{x}_5 \qquad + x_5 &&= 0
\end{aligned}
$$

The reference constraint is

$$x_1 + x_2 + x_3 + x_4 + x_5 + y_5 = 11$$

where $a_{ro} = 11$ was determined by solving the linear program as described in Section 3. The first tableau is:

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | b | |
|---|---|---|---|---|---|---|
| 2 | 3 | 1 | 2 | 2 | 18 | |
| 3 | 2 | 2 | 1 | 2 | 15 | |
| -6 | 0 | 1 | 0 | 0 | 15 | |
| 0 | -7 | 0 | 1 | 0 | 0 | |
| -1 | -7 | 0 | 0 | 0 | 0 | |
| 0 | -1 | 0 | 0 | 0 | 0 | |
| 0 | 0 | -1 | 0 | 0 | 0 | |
| 0 | 0 | 0 | -1 | 0 | 0 | |
| 0 | 0 | 0 | 0 | -1 | 0 | |
| 1 | 1 | 1 | 1 | 1 | 11 | reference constraint |
| 0 | 0 | -1 | -1 | -1 | 0 | objective function |

where the identity matrix has been omitted.

A transition cycle can be executed immediately by selecting column 5 as the pivot column. The minimum ratio $\theta_5 = 7\frac{1}{2}$ and the source row is row 2. A pivot is carried out using rule 5a of the algorithm. The next tableau is:

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 1 | -1 | 2 | -2 | 4 | |
| 1 | 0 | 0 | 1 | -2 | 1 | |
| -6 | 0 | 1 | 0 | 0 | 0 | |
| 0 | -7 | 0 | 1 | 0 | 0 | |
| -1 | 0 | 0 | 0 | 0 | 0 | |
| 0 | -1 | 0 | 0 | 0 | 0 | Tableau 1 |
| 0 | 0 | -1 | 0 | 0 | 0 | |
| 0 | 0 | 0 | -1 | 0 | 0 | |
| 1 | 1 | 1 | 0 | 1 | 7 | |
| 0 | 0 | 0 | 1 | -1 | 4 | RC |
| 1 | 1 | 0 | -1 | 1 | 7 | OF |

Since a transition cycle was executed, the reference constraint must be modified to:

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 11 |

The problem is acutally at optimality now but it does not meet the criteria for optimality yet. The pivot column for the next iteration is column 4 (the only column eligible). $\theta_4 = 0$ and the only row in S(4) is row 4. A stationary cycle must be executed. The next tableau is:

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 15 | -1 | -2 | -2 | 4 | |
| 1 | 7 | 0 | -1 | -2 | 1 | |
| -6 | 0 | 1 | 0 | 0 | 0 | |
| -1 | 0 | 0 | 0 | 0 | 0 | |
| 0 | -1 | 0 | 0 | 0 | 0 | Tableau 2 |
| 0 | 0 | -1 | 0 | 0 | 0 | |
| 0 | -7 | 0 | 1 | 0 | 0 | |
| 1 | 1 | 1 | 0 | 1 | 7 | |
| 1 | 8 | 1 | -1 | 1 | 11 | |
| 1 | -6 | 0 | 1 | 1 | 7 | |

Again only one column is available as the pivot column, column 2. $\theta_2 = 1/7$ and $S(2) = (1,2)$. Using the criteria of SRR3 the two G vectors are:

$G(1) = (0,1,-1,-1-1,0)$

$G(2) = (0,1,0,-1,-1,0)$

therefore row 2 is selected as the source row. Again a stationary cycle must be executed. The next tableau is:

| | | | | | |
|---|---|---|---|---|---|
| 0 | -15 | -1 | 13 | 13 | 4 |
| 1 | -7 | 0 | 6 | 5 | 1 |
| 0 | 0 | 0 | -1 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | -1 | -1 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | 7 | 0 | -6 | -7 | 0 |
| 1 | -1 | 1 | 1 | 2 | 7 |
| 1 | -8 | 1 | 7 | 9 | 11 |
| 1 | 6 | 0 | -5 | -5 | 7 |

Tableau 3

Two columns are eligible as pivot columns, 4 and 5. The first components of the reference vectors are:

$\bar{A}(4) = (-5/7,...)$

$\bar{A}(5) = (-5/9,...)$

Since $\bar{A}(4) \overset{L}{<} \bar{A}(5)$, column 4 is selected as the pivot column. $\theta_4 = 1/6$ and $S(4) = (1,2)$. Row 2 was the source row for the last tableau so it will be selected as source row as long as it is in $S(v)$. The next tableau is:

| | | | | | |
|---|---|---|---|---|---|
| 0 | 11 | -1 | -13 | 13 | 4 |
| 1 | 5 | 0 | -6 | 5 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | -2 | 0 | 1 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | -1 | 0 | 1 | -1 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | -5 | 0 | 6 | -7 | 0 |
| 1 | 1 | 1 | -1 | 2 | 7 |
| 1 | 6 | 1 | -7 | 9 | 11 |
| 1 | -4 | 0 | 5 | -5 | 7 |

Tableau 4

All aspects of the algorithm have been covered in the first four tableaus. The following tableaus will be presented without further comment. The pivot element will be listed below each tableau. The pivot element for the fourth tableau is (2,2)

Tableau 5

| | | | | | |
|---|---|---|---|---|---|
| 0 | -11 | -1 | 9 | 2 | 4 |
| 1 | -5 | 0 | 4 | 0 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | 2 | 0 | -3 | 2 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | -1 | 0 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | 5 | 0 | -4 | -2 | 0 |
| 1 | -1 | 1 | 1 | 1 | 7 |
| 1 | -6 | 1 | 5 | 3 | 11 |
| 1 | 4 | 0 | -3 | -1 | 7 |

Pivot Element = (2,4)

Tableau 6

| | | | | | |
|---|---|---|---|---|---|
| 0 | 7 | -1 | -9 | 2 | 4 |
| 1 | 3 | 0 | -4 | 0 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | -4 | 0 | 3 | 2 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | -1 | 0 | 1 | 0 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | -3 | 0 | 4 | -2 | 0 |
| 1 | 1 | 1 | -1 | 1 | 7 |
| 1 | 4 | 1 | -5 | 3 | 11 |
| 1 | -2 | 0 | 3 | -1 | 7 |

Pivot Element = (2,2)

Tableau 7

| | | | | | |
|---|---|---|---|---|---|
| 0 | -7 | -1 | 5 | 2 | 4 |
| 1 | -3 | 0 | 2 | 0 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | 4 | 0 | -5 | 2 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | -1 | 0 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | 3 | 0 | -2 | -2 | 0 |
| 1 | -1 | 1 | 1 | 1 | 7 |
| 1 | -4 | 1 | 3 | 3 | 11 |
| 1 | 2 | 0 | -1 | -1 | 7 |

Pivot Element = (4,5)

Tableau 8

| | | | | | |
|---|---|---|---|---|---|
| 0 | -11 | -1 | 11 | -2 | 4 |
| 1 | -3 | 0 | 2 | 0 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | -2 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | -1 | 0 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | 7 | 0 | -8 | 2 | 0 |
| 1 | -3 | 1 | 4 | -1 | 7 |
| 1 | -10 | 1 | 12 | -3 | 11 |
| 1 | 4 | 0 | -4 | 1 | 7 |

Pivot Element = (4,4)

Tableau 9

| | | | | | |
|---|---|---|---|---|---|
| 0 | -11 | -1 | -11 | 20 | 4 |
| 1 | -3 | 0 | -2 | 4 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | -1 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | -2 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | 7 | 0 | 8 | -14 | 0 |
| 1 | -3 | 1 | -4 | 7 | 7 |
| 1 | -10 | 1 | -12 | 21 | 11 |
| 1 | 4 | 0 | 4 | -7 | 7 |

Pivot Element = (10,5)

Tableau 10

| | | | | | |
|---|---|---|---|---|---|
| 0 | 9 | -1 | 9 | -20 | 4 |
| 1 | 1 | 0 | 2 | -4 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | -1 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | -1 | 0 | -1 | 2 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | -7 | 0 | -6 | 14 | 0 |
| 1 | 4 | 1 | 3 | -7 | 7 |
| 1 | 11 | 1 | 9 | -21 | 11 |
| 1 | -3 | 0 | -3 | 7 | 7 |

Pivot Element = (1,4)

### Tableau 11

| | | | | | |
|---|---|---|---|---|---|
| 0 | 0 | 8 | -9 | 7 | 4 |
| 1 | -1 | 2 | -2 | 2 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | -1 | 1 | -3 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | -1 | 1 | -1 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | -1 | -6 | 6 | -4 | 0 |
| 1 | 1 | 4 | -3 | 2 | 7 |
| 1 | 2 | 10 | -9 | 6 | 11 |
| 1 | 0 | -3 | 3 | -2 | 7 |

Pivot Element = (1,5)

### Tableau 12

| | | | | | |
|---|---|---|---|---|---|
| 0 | 0 | 1 | 5 | -7 | 4 |
| 1 | -1 | 0 | 2 | -2 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 2 | -5 | 3 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | -1 | 1 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | -1 | -2 | -2 | 4 | 0 |
| 1 | 1 | 2 | 1 | -2 | 7 |
| 1 | 2 | 4 | 3 | -6 | 11 |
| 1 | 0 | -1 | -1 | 2 | 7 |

Pivot Element = (1,4)

### Tableau 13

| | | | | | |
|---|---|---|---|---|---|
| 0 | 0 | 1 | -5 | 3 | 4 |
| 1 | -1 | 0 | -2 | 2 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 2 | 5 | -7 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | -1 | 0 |
| 0 | 0 | -1 | 0 | 0 | 0 |
| 0 | -1 | -2 | 2 | 0 | 0 |
| 1 | 1 | 2 | -1 | 0 | 7 |
| 1 | 2 | 4 | -3 | 0 | 11 |
| 1 | 0 | -1 | 1 | 0 | 7 |

Pivot Element = (4,3)

### Tableau 14

| | | | | | |
|---|---|---|---|---|---|
| 0 | 0 | -1 | -7 | 7 | 4 |
| 1 | -1 | 0 | -2 | 2 | 1 |
| -6 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | -2 | 1 | 1 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | -1 | 0 |
| 0 | 0 | 1 | 2 | -4 | 0 |
| 0 | -1 | 2 | 6 | -8 | 0 |
| 1 | 1 | -2 | -5 | 8 | 7 |
| 1 | 2 | -4 | -11 | 16 | 11 |
| 1 | 0 | 1 | 3 | -4 | 7 |

Pivot Element = (4,5)

### Tableau 15

| | | | | | |
|---|---|---|---|---|---|
| 0 | -7 | 13 | -14 | -7 | 4 |
| 1 | -3 | 4 | -4 | -2 | 1 |
| -6 | -4 | 7 | -6 | -4 | 0 |
| 0 | 0 | 0 | 0 | -1 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | -2 | 2 | 1 | 0 |
| 0 | 4 | -7 | 6 | 4 | 0 |
| 0 | 7 | -14 | 14 | 8 | 0 |
| 1 | -7 | 14 | -13 | -8 | 7 |
| 1 | -14 | 28 | -27 | -16 | 11 |
| 1 | 4 | -7 | 7 | 4 | 7 |

Pivot Element = (10,3)

### Tableau 16

| | | | | | |
|---|---|---|---|---|---|
| 0 | 6 | -13 | -1 | 6 | 4 |
| 1 | 1 | -4 | 0 | 2 | 1 |
| -6 | 3 | -7 | 1 | 3 | 0 |
| 0 | 0 | 0 | 0 | -1 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | -1 | 2 | 0 | -1 | 0 |
| 0 | -3 | 7 | -1 | -3 | 0 |
| 0 | -7 | 14 | 0 | -6 | 0 |
| 1 | 7 | -14 | 1 | 6 | 7 |
| 1 | 14 | -28 | 1 | 12 | 11 |
| 1 | -3 | 7 | 0 | -3 | 7 |

Pivot Element = (10,5)

Tableau 17

| | | | | | |
|---|---|---|---|---|---|
| 0 | 0 | 5 | -1 | -6 | 4 |
| 1 | -1 | 2 | 0 | -2 | 1 |
| -6 | 0 | 2 | 1 | -3 | 0 |
| 0 | 1 | -3 | 0 | 1 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | -1 | 0 | 1 | 0 |
| 0 | 0 | -2 | -1 | 3 | 0 |
| 0 | -1 | -4 | 0 | 6 | 0 |
| 1 | 1 | 4 | 1 | -6 | 7 |
| 1 | 2 | 8 | 1 | -12 | 11 |
| 1 | 0 | -2 | 0 | 3 | 7 |

Pivot Element = (1,3)

Tableau 18

| | | | | | |
|---|---|---|---|---|---|
| 0 | 0 | -5 | 4 | 4 | 4 |
| 1 | -1 | -2 | 2 | 2 | 1 |
| -6 | 0 | -2 | 3 | 1 | 0 |
| 0 | 1 | 3 | -3 | -5 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | -1 | -1 | 0 |
| 0 | 0 | 2 | -3 | -1 | 0 |
| 0 | -1 | 4 | -4 | -2 | 0 |
| 1 | 1 | -4 | 5 | 2 | 7 |
| 1 | 2 | -8 | 9 | 4 | 11 |
| 1 | 0 | 2 | -2 | -1 | 7 |

Pivot Element = (2,5)

Tableau 19

| | | | | | |
|---|---|---|---|---|---|
| 0 | 4 | -1 | 0 | -4 | 4 |
| 1 | 1 | 0 | 0 | -2 | 1 |
| -6 | 1 | -1 | 2 | -1 | 0 |
| 0 | -4 | -2 | 2 | 5 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | -1 | 0 | 0 | 1 | 0 |
| 0 | -1 | 1 | -2 | 1 | 0 |
| 0 | -3 | 2 | -2 | 2 | 0 |
| 1 | 3 | -2 | 3 | -2 | 7 |
| 1 | 6 | -4 | 5 | -4 | 11 |
| 1 | -1 | 1 | -1 | 1 | 7 |

Pivot Element = (4,4)

Tableau 20

| | | | | | |
|---|---|---|---|---|---|
| 0 | 4 | -1 | 0 | -4 | 4 |
| 1 | 1 | 0 | 0 | -2 | 1 |
| -6 | 5 | 1 | -2 | -5 | 0 |
| 0 | 0 | 0 | -2 | 1 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | -1 | 0 | 0 | 1 | 0 |
| 0 | -5 | -1 | 2 | 5 | 0 |
| 0 | -7 | 0 | 2 | 6 | 0 |
| 1 | 9 | 1 | -3 | -8 | 7 |
| 1 | 16 | 1 | -5 | -14 | 11 |
| 1 | -3 | 0 | 1 | 3 | 7 |

Pivot Element = (10,2)

Tableau 21

| | | | | | |
|---|---|---|---|---|---|
| 0 | -4 | -1 | 4 | 0 | 4 |
| 1 | -1 | 0 | 1 | -1 | 1 |
| -6 | -5 | 1 | 3 | 0 | 0 |
| 0 | 0 | 0 | -2 | 1 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | -1 | 0 | 0 |
| 0 | 5 | -1 | -3 | 0 | 0 |
| 0 | 7 | 0 | -5 | -1 | 0 |
| 1 | -9 | 1 | 6 | 1 | 7 |
| 1 | -16 | 1 | 11 | 2 | 11 |
| 1 | 3 | 0 | -2 | 0 | 7 |

Pivot Element = (3,4)

Tableau 22

| | | | | | |
|---|---|---|---|---|---|
| 8 | 4 | -1 | -4 | 0 | 4 |
| 3 | 1 | 0 | -1 | -1 | 1 |
| 0 | 1 | 1 | -3 | 0 | 0 |
| -4 | -4 | 0 | 2 | 1 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| -2 | -1 | 0 | 1 | 0 | 0 |
| -6 | -1 | -1 | 3 | 0 | 0 |
| -10 | -3 | 0 | 5 | -1 | 0 |
| 13 | 3 | 1 | -6 | 1 | 7 |
| 23 | 6 | 1 | -11 | 2 | 11 |
| -3 | -1 | 0 | 2 | 0 | 7 |

Pivot Element = (3,2)

Tableau 23

| | | | | | |
|---|---|---|---|---|---|
| 8 | -4 | -5 | 8 | 0 | 4 |
| 3 | -1 | -1 | 2 | -1 | 1 |
| 0 | -1 | 0 | 0 | 0 | 0 |
| -4 | 4 | 4 | -10 | 1 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| -2 | 1 | 1 | -2 | 0 | 0 |
| -6 | 1 | 0 | 0 | 0 | 0 |
| -10 | 3 | 3 | -4 | -1 | 0 |
| 13 | -3 | -2 | 3 | 1 | 7 |
| 23 | -6 | -5 | 7 | 2 | 11 |
| -3 | 1 | 1 | -1 | 0 | 7 |

Pivot Element = (1,4)

Tableau 24

| | | | | | |
|---|---|---|---|---|---|
| 0 | 4 | 3 | -8 | 0 | 4 |
| 1 | 1 | 1 | -2 | -1 | 1 |
| 0 | -1 | 0 | 0 | 0 | 0 |
| 6 | -6 | -6 | 10 | 1 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 |
| 0 | -1 | -1 | 2 | 0 | 0 |
| -6 | 1 | 0 | 0 | 0 | 0 |
| -6 | -1 | -1 | 4 | -1 | 0 |
| 10 | 0 | 1 | -3 | 1 | 7 |
| 16 | 1 | 2 | -7 | 2 | 11 |
| -2 | 0 | 0 | 1 | 0 | 7 |

Pivot Element = (10,1)

Tableau 25

| | | | | | |
|---|---|---|---|---|---|
| 0 | 4 | 3 | -8 | 0 | 4 |
| -1 | 1 | 1 | -1 | -1 | 1 |
| 0 | -1 | 0 | 0 | 0 | 0 |
| -6 | -6 | -6 | 16 | 1 | 0 |
| 1 | 0 | 0 | -1 | 0 | 0 |
| 0 | -1 | -1 | 2 | 0 | 0 |
| 6 | 1 | 0 | -6 | 0 | 0 |
| 6 | -1 | -1 | -2 | -1 | 0 |
| -10 | 0 | 1 | 7 | 1 | 7 |
| -16 | 1 | 2 | 9 | 2 | 11 |
| 2 | 0 | 0 | -1 | 0 | 7 |

Pivot Element = (6,4)

Tableau 26

| | | | | | |
|---|---|---|---|---|---|
| 0 | -4 | -5 | 8 | 0 | 4 |
| -1 | 0 | 0 | 1 | -1 | 1 |
| 0 | -1 | 0 | 0 | 0 | 0 |
| -6 | 10 | 10 | -16 | 1 | 0 |
| 1 | -1 | -1 | 1 | 0 | 0 |
| 0 | 1 | 1 | -2 | 0 | 0 |
| 6 | -5 | -6 | 6 | 0 | 0 |
| 6 | -3 | -3 | 2 | -1 | 0 |
| -10 | 7 | 8 | -7 | 1 | 7 |
| -16 | 10 | 11 | -9 | 2 | 11 |
| 2 | -1 | -1 | 1 | 0 | 7 |

Pivot Element = (6,2)

Tableau 27

| | | | | | |
|---|---|---|---|---|---|
| 0 | 4 | -1 | 0 | 0 | 4 |
| -1 | 0 | 0 | 1 | -1 | 1 |
| 0 | 1 | 1 | -2 | 0 | 0 |
| -6 | -10 | 0 | 4 | 1 | 0 |
| 1 | 1 | 0 | -1 | 0 | 0 |
| 0 | -1 | 0 | 0 | 0 | 0 |
| 6 | 5 | -1 | -4 | -1 | 0 |
| 6 | 3 | 0 | -4 | -1 | 0 |
| -10 | -7 | 1 | 7 | 1 | 7 |
| -16 | -10 | 1 | 11 | 2 | 11 |
| 2 | 1 | 0 | -1 | 0 | 7 |

Pivot Element = (4,4)

Tableau 28

| | | | | | |
|---|---|---|---|---|---|
| 0 | 4 | -1 | 0 | 0 | 4 |
| 1 | 3 | 0 | -1 | -1 | 1 |
| -4 | -5 | 1 | 2 | 0 | 0 |
| 2 | 2 | 0 | -4 | 1 | 0 |
| -1 | -2 | 0 | 1 | 0 | 0 |
| 0 | -1 | 0 | 0 | 0 | 0 |
| -2 | -7 | -1 | 4 | 0 | 0 |
| -2 | -9 | 0 | 4 | -1 | 0 |
| 4 | 14 | 1 | -7 | 1 | 7 |
| 6 | 23 | 1 | -11 | 2 | 11 |
| 0 | -2 | 0 | 1 | 0 | 7 |

Final Tableau

Tableau 28 meets the optimality criteria of $a_{ov}/a_{rv} > -1/a_{ro}$, that is $-2/23 > -1/11$, although the problem is not dual feasible.

The optimal values of the variables are:

$x_o = 7$

$x_1 = 0$

$x_2 = 0$

$x_3 = 0$

$x_4 = 0$

$x_5 = 7$

and the values of the slack variables are:

$y_1 = 4$

$y_2 = 1$

$y_3 = 0$

$y_4 = 0$

## Appendix B

```
      INTEGER T(50,50)/2500*0/
      INTEGER G(50),GM(50)
      DIMENSION R(50),RM(50)
      ID=50
      JD=50
C
C          DIMENSIONED VARIABLES
C              T-THE TABLEAU MATRIX
C              G&GM- USED IN PIVELM SUBROUTINE
C              TO DETERMINE THE SOURCE ROW
C              R&RM- THE REFERENCE VECTOR
C
C          READ IN THE FOLLOWING DATA
C              NP1- NUMBER OF FIRST PROBLEM
C              NP2- NUMBER OF LAST PROBLEM
C              INX- DETERMINES WHAT TABLEAUS
C              ARE TO BE PRINTED OUT
C              0-FIRST AND LAST ONLY
C              1-ALL TABLEAUS
C              N>1 EVERY N-TH TABLEAU
C
      READ(5,100) NP1,NP2,INX
      DO 23  III=NP1,NP2
      WRITE(6,800) III
  800 FORMAT(1H1, 20X,'  PROBLEM  ',I4,//)
      READ(5,100)M,N,NL,NE,NG,LMAX
  100 FORMAT(10I8)
C
C          READ IN THE FOLLOWING DATA
C              M-NUMBER OF CONSTRAINTS
C              N- NUMBER OF VARIABLES
C              NL- NUMBER OF LESS THAN CONSTRAINTS
C              NE-NUMBER OF EQUALITY CONSTRAINTS
C              NG-NUMBER OF GREATER  THAN
C              CONSTRAINTS
C              LMAX-THE UPPER LIMIT ON THE
C              SUMMATION OF THE VARIABLES
C
      KK=1
      MN1=M+N+NG+2
      MN=MN1-2
      L=MN1-1
      MN11=MN1+1
      N1=N+NG+1
      M1=M+1
      NN=N1-1
      NX=1
      IF(III.EQ.NP1) GO TO 554
      DO 553 I=1,MN1
      DO 553 J=1,N1
      T(I,J)=0
  553 CONTINUE
C
C          READ IN A MATRIX AND B VECTOR
C            NOTE-ORDER OF CONSTRAINTS IN THE A
C            MATRIX MUST BE ALL LESS THANS,
C            ALL EQUALITY AND THEN ALL GREATER
C            THAN CONSTRAINTS
C
  554 DO 10 I=1,M
      READ(5,100)(T(I,J),J=1,N),T(I,N1)
   10 CONTINUE
C
C          ADD SURPLUS VARIABLES
C
```

```
      IF(NG.EQ.0) GO TO 121
      K=1
      I1=NL+NE+1
      DO 12 I=I1,M
      T(I,N+K)=-1
      K=K+1
   12 CONTINUE
C
C                READ IN COST VECTOR
C
  121 READ(5,100) (T(MN1,J),J=1,N)
      K=1
      DO 13 I=M1,MN
      T(I,K)=-1
      T(MN1,K)=-T(MN1,K)
      K=K+1
   13 CONTINUE
      IF(NE+NG.EQ.0) GO TO 131
C
C                DETERMINE W EQUATION FOR PHASE ONE
C
      I1=NL+1
      CALL SCIM(T,I1,M,1,N1,MN11,-1,ID,JD)
  131 DO 50 J=1,NN
      T(L,J)=1
   50 CONTINUE
      T(L,N1)=LMAX
      I=1
      NT=1
      CALL OUT1 (T,MN11,N1,NT,ID,JD)
      IF(NE+NG.EQ.0) GO TO 126
C
C         PHASE ONE
C
   25 CALL PIVELM(T,I,J,N1,MN,MN11,NN,L,ID,JD,G,GM,R,RM
     * ,IX)
      IF(T(MN11,N1).EQ.0) GO TO 44
   43 IF(J.NE.0) GO TO 16
      WRITE(6,225)
  225 FORMAT(//,20X,' NO BASIC FEASIBLE SOLUTION ',////
     * )
      GO TO 23
   16 CALL NXTTAB (T,MN11,N1,I,J,ID,JD)
      CALL ELIM (T,NL,M,N1,MN1,NN,MN,MN11,L,ID,JD)
   62 IF(NT.NE.KK*INX) GO TO 41
      KK=KK+1
      WRITE(6,203)
  203 FORMAT(////)
      CALL OUT1 (T,MN11,N1,NT,ID,JD)
   41 NT=NT+1
      IF(IX.NE.1) GO TO 25
      DO 51 J=1,NN
      T(L,J)=1
   51 CONTINUE
      T(L,N1)=LMAX
      GO TO 25
C
C                ELIMINATE ARTIFICIALS  STILL PRESENT
C                WHEN W=0
C
   44 IF(M.EQ.NL) GO TO 125
      CALL MODPIV(T,ID,JD,I,J,M,MN11,NN)
      CALL NXTTAB (T,MN11,N1,I,J,ID,JD)
      CALL ELIM (T,NL,M,N1,MN1,NN,MN,MN11,L,ID,JD)
      CALL OUT1(T,MN11,N1,NT,ID,JD)
      GO TO 44
C
C
C         PHASE TWO
C
C
  125 DO 52 J=1,NN
      T(L,J)=1
   52 CONTINUE
      T(L,N1)=LMAX
  126 CALL PIVELM(T,I,J,N1,MN,MN1 ,NN,L,ID,JD,G,GM,R,RM
     * ,IX)
      IF(J.EQ.0) GO TO 26
      IF(T(L,N1).EQ.0) GO TO 26
      IF(FLOAT(T(MN1,J))/FLOAT(T(L,J)).GT.(-1)/FLOAT(T(
     * L,N1))) GO TO 26
      CALL NXTTAB (T,MN1,N1,I,J,ID,JD)
      IF(NT.NE.KK*INX) GO TO 42
      KK=KK+1
      WRITE(6,203)
      CALL OUT1 (T,MN1 ,N1,NT,ID,JD)
   42 NT=NT+1
      IF(IX.NE.1) GO TO 126
      DO 53 J=1,NN
      T(L,J)=1
   53 CONTINUE
      T(L,N1)=LMAX
      GO TO 126
C
C              WRITE OUT FINAL TABLEAU AND ANSWERS
C
   26 WRITE(6,800) III
      WRITE(6,201)
  201 FORMAT(30X,' FINAL TABLEAU ')
      CALL OUT1 (T,MN1 ,N1,NT,ID,JD)
      CALL OUT2 (T,M,N,NL,NE,NG,MN1,N1,NT,ID,JD)
   23 CONTINUE
     END


      SUBROUTINE ELIM (T,NL,M,N1,MN1,NN,MN,MN11,L,ID,JD
     * )
      INTEGER T(ID,JD)
C
C              THIS SUBROUTINE ELIMINATES THE ARTIFICAL
C              VARIABLES FROM THE PROBLEM
C
      L1=NL+1
      KK=0
      DO 10 LL=L1,M
      I=LL-KK
      K=0
      DO 11 J=1,N1
      IF(T(I,J).EQ.0.OR.T(I,J).EQ.-1) GO TO 12
      GO TO 10
   12 IF(T(I,J).NE.-1) GO TO 11
      K=K+1
      J1=J
   11 CONTINUE
      IF(K.NE.1) GO TO 10
      DO 31 JJ=J1,N1
      DO 31 II=1,MN11
      T(II,JJ)=T(II,JJ+1)
   31 CONTINUE
```

```
      NN=NN-1
      N1=N1-1
      DO 32 II=I,MN1
      DO 32 JJ=1,N1
      T(II,JJ)=T(II+1,JJ)
   32 CONTINUE
      MN11=MN1
      MN1=MN1-1
      MN=MN-1
      L=L-1
      KK=KK+1
   10 CONTINUE
      M=M-KK
      RETURN
      END


      SUBROUTINE OUT1(T,M,N,IX,ID,JD)
      INTEGER T(ID,JD)
C
C               THIS ROUTINE WRITES OUT THE TABLEAUS
C
      WRITE(6,102) IX
      DO 10 I=1,M
      WRITE(6,100)(T(I,J),J=1,N)
      WRITE(6,101)
   10 CONTINUE
  100 FORMAT(12(1X,I9))
  101 FORMAT(//)
 102 FORMAT(//,10X,' TABLEAU ',I9,//)
  103 FORMAT(20(1X,F5.1))
     RETURN
     END


      SUBROUTINE OUT2 (T,M,N,NL,NE,NG,MN1,N1,NT,ID,JD)
      INTEGER T(ID,JD)
C
C               THIS ROUTINE WRITES OUT THE FINAL ANSWERS
C
      WRITE(6,202)
  202 FORMAT(////)
      IF(NL.EQ.0) GO TO 11
      DO 10 I=1,NL
      WRITE(6,101)I,T(I,N1)
  101 FORMAT(//,20X,' SLACK ',I4,' = ',I8,//)
   10 CONTINUE
   11 IF(NG.EQ.0) GO TO 13
      I1=MN1-NG
      I2=MN1-1
      K=1
      DO 12 I=I1,I2
      WRITE(6,204) K,T(I,N1)
  204 FORMAT(//,20X,' SURPLUS ',I3,' =',I8,/)
      K=K+1
   12 CONTINUE
   13 I1=M+1
      I2=M+N
      K=1
      DO 14 I=I1,I2
      WRITE(6,102) K,T(I,N1)
  102 FORMAT(//,20X,' X ',I4,' = ',I8,//)
      K=K+1
   14 CONTINUE
      WRITE(6,103) T(MN1,N1)
  103 FORMAT(//,20X,' Z = ',I8,//)
      RETURN
      END


      FUNCTION IDIV(L,M)
C
C         THIS FUNCTION DETERMINES THE GREATEST IVTEGER L
     * ESS THAN L/M
C                NOTE-M MUST BE GREATER THAN 0
C
      IF(M.LE.0) GO TO 98
      IF(L)1,2,3
    1 K=-1
      GO TO 4
    2 IDIV=0
      GO TO 99
    3 K=1
    4 LA=K*L
      N=0
   13 LA=LA-M
      IF(LA.LE.0) GO TO 14
      N=N+1
      GO TO 13
   14 IF(K.LT.0.OR.LA.EQ.0) GO TO 16
      IDIV=N
      GO TO 99
   16 IDIV=K*(N+1)
      GO TO 99
   98 WRITE(6,100)
  100 FORMAT(//,10X,'  ERROR IN IDIV   M<=0 ',//)
      IDIV=0
   99 RETURN
      END


      SUBROUTINE NXTTAB (T,M,N,I,J,ID,JD)
      INTEGER T(ID,JD)
      INTEGER G
C
C                THIS ROUTINE CARRIES OUT THE PIVOT
C
      DO 11 K=1,N
      IF(K.EQ.J) GO TO 11
      G=IDIV(T(I,K),T(I,J))
   14 DO 11 L=1,M
      T(L,K)=T(L,K)-T(L,J)*G
   11 CONTINUE
      DO 12 L=1,M
      T(L,J)=-T(L,J)
   12 CONTINUE
      RETURN
      END


      SUBROUTINE SCIM (T,I1,I2,J1,J2,L,IX,ID,JD)
      INTEGER T(ID,JD)
C
C                THIS ROUTINE SUMS THE COLUMNS OF AN INTEGE
     * R MATRIX
```

```
C
      DO 10 J=J1,J2
      ISUM=0
      DO 11 I=I1,I2
      ISUM=ISUM+T(I,J)
   11 CONTINUE
      T(L,J)=ISUM*IX
   10 CONTINUE
      RETURN
      END


      SUBROUTINE MODPIV(T,ID,JD,IM,JM,M,MN11,NN)
      INTEGER T(ID,JD)
      DO 10 J=1,NN
      IF(T(MN11,J).GT.0) GO TO 10
      DO 10 I=1,M
      IF(T(I,J).EQ.1) GO TO 12
      IF(T(I,J).LE.0) GO TO 10
      IM=I
      JM=J
   10 CONTINUE
      RETURN
   12 IM=I
      JM=J
      RETURN
      END


      SUBROUTINE PIVELM (T,IM,JM,N1,MN,MN1,NN,L,ID,JD,G
     * ,GM,R,RM,IX)
      INTEGER T(ID,JD),G(JD),GM(JD)
      DIMENSION R(ID),RM(ID)
C
C              THIS ROUTINE DETERMINES THE PIVOT ELEMENT
C
      IRM=-9
      JM=0
      JJ=0
      DO 10 J=1,NN
      IF(T(MN1,J).GE.0) GO TO 10
      IRT=999999
      DO 12 I=1,MN
      IF(T(I,J).LE.0) GO TO 12
      IR=T(I,N1)/T(I,J)
      IF(IR.GE.IRT) GO TO 12
      IRT=IR
      LL=I
   12 CONTINUE
      IF(IRT.GE.999999.OR.IRT.LE.IRM) GO TO 10
      JJ=J
      II=LL
      IRM=IRT
   10 CONTINUE
      IF(JJ.EQ.0.OR.IRM.GE.1) GO TO 9
      GO TO 8
    9 IM=II
      JM=JJ
      IX=1
      GO TO 23
C
C              DETERMINE PIVOT COLUMN FOR STATIONARY CYCL
     * E
```

```
C
    8 DO 11 I=1,MN1
      R(I)=0
      RM(I)=999999
   11 CONTINUE
      DO 13 J=1,NN
      IF(T(MN1,J).GE.0.OR.T(L,J).LE.0) GO TO 13
      R(1)=FLOAT(T(MN1 ,J))/FLOAT(T(L,J))
      DO 14 I=2,MN1
      R(I)=FLOAT(T(I-1,J))/FLOAT(T(L,J))
   14 CONTINUE
    7 DO 15 I=1,MN
      IF(R(I).LT.RM(I)) GO TO 16
      IF(RM(I).LT.R(I)) GO TO 13
   15 CONTINUE
      GO TO 13
   16 DO 17 I=1,MN
      RM(I)=R(I)
   17 CONTINUE
      JM=J
   13 CONTINUE
      IF(JM.EQ.0) GO TO 22
C
C              DETERMINE SOURCE ROW
C
      IF(T(IM ,JM).GT.T(IM ,N1)) GO TO 23
      IF(T(L,JM).LE.T(L,N1)) GO TO 18
      IM=L
      IX=4
      GO TO 23
   18 DO 19 J=1,NN
      GM(J)=-9999999
   19 CONTINUE
      DO 40 I=1,L
      IF(T(I,JM).LE.T(I,N1)) GO TO 40
      DO 41 J=1,NN
      G(J)=IDIV(T(I,J),T(I,JM))
   41 CONTINUE
      DO 42 J=1,NN
      IF(G(J).LT.GM(J)) GO TO 40
      IF(G(J).GT.GM(J)) GO TO 43
   42 CONTINUE
      GO TO 40
   43 DO 44 J=1,NN
      GM(J)=G(J)
   44 CONTINUE
      IM=I
   40 CONTINUE
      IX=2
      GO TO 23
   22 IX=3
      JM=0
      WRITE(6,100)
   23 RETURN
   34 WRITE(6,101)IM,JM,IX
      WRITE(6,200) (RM(I),I=1,MN)
  100 FORMAT(//,20X,' ERROR IN PIVELM ',//)
  101 FORMAT(//,20X,' PIVELM = ',5I3,//)
  200 FORMAT(//,10X,10F11.4,//)
      RETURN
      END
```

# INITIAL DISTRIBUTION LIST

| | | No. Copies |
|---|---|---|
| 1. | Defense Documentation Center<br>Cameron Station<br>Alexandria, Virginia 22314 | 20 |
| 2. | Library<br>Naval Postgraduate School<br>Monterey, California 93940 | 2 |
| 3. | Chief of Naval Operations (OP-96)<br>Department of the Navy<br>Washington, D.C. 20350 | 1 |
| 4. | Professor Harold Greenberg<br>Department of Operations Analysis<br>Naval Postgraduate School<br>Monterey, California 93940 | 1 |
| 5. | LT Howard T. Schottle USN<br>937 Rosita Road<br>Del Rey Oaks, California 93940 | 1 |
| 6. | Operations Analysis Department<br>Naval Postgraduate School<br>Monterey, California 93940 | 1 |

UNCLASSIFIED
Security Classification

## DOCUMENT CONTROL DATA - R & D

*Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)*

| | |
|---|---|
| 1. ORIGINATING ACTIVITY (*Corporate author*)<br>Naval Postgraduate School<br>Monterey, California 93940 | 2a. REPORT SECURITY CLASSIFICATION<br>UNCLASSIFIED<br>2b. GROUP |

3. REPORT TITLE

ON PRIMAL INTEGER PROGRAMMING METHODS

4. DESCRIPTIVE NOTES (*Type of report and inclusive dates*)
M.S. Thesis - Operations Research, 1968

5. AUTHOR(S) (*First name, middle initial, last name*)

Howard T. SCHOTTLE

| | | |
|---|---|---|
| 6. REPORT DATE<br>December 1968 | 7a. TOTAL NO. OF PAGES<br>44 | 7b. NO. OF REFS<br>10 |
| 8a. CONTRACT OR GRANT NO.<br>b. PROJECT NO.<br>c.<br>d. | 9a. ORIGINATOR'S REPORT NUMBER(S)<br>9b. OTHER REPORT NO(S) (*Any other numbers that may be assigned this report*) | |

10. DISTRIBUTION STATEMENT

Distribution of this document is unlimited.

| | |
|---|---|
| 11. SUPPLEMENTARY NOTES | 12. SPONSORING MILITARY ACTIVITY |

13. ABSTRACT

Advances in the field of integer programming have recently been made by Young and Glover, who have developed primal all integer integer programming algorithms. It is the purpose of this paper to gain computational experience and suggest techniques to improve the efficiency of the algorithm. A first phase procedure is developed to obtain an initial feasible integer solution.

DD FORM 1 NOV 65 1473 (PAGE 1)
S/N 0101-807-6811

UNCLASSIFIED
Security Classification

A-31408

| 14 KEY WORDS | LINK A | | LINK B | | LINK C | |
|---|---|---|---|---|---|---|
| | ROLE | WT | ROLE | WT | ROLE | WT |
| Programming | | | | | | |
| Integer Programming | | | | | | |
| Linear Integer Programming | | | | | | |
| Primal Integer Programming | | | | | | |

1 AUG 69
28 OCT 69

S10069
17372

107537

Thesis
S342
c.2

Schottle
On primal integer
programming methods.

1 AUG 69
28 OCT 69
28 OCT 69

S10069
17372

107537

Thesis
S342
c.2

Schottle
On primal integer
programming methods.

thesS342
On primal integer programming methods.
3 2768 001 00334 6
DUDLEY KNOX LIBRARY